Stanford Exploration Project, Report 72, December 18, 1997, pages 1–42

Short Note

Eigenvector formulation for missing data

Jon F. Claerbout[1]

keywords: *Fourier transform, interpolation, deconvolution, linear, missing data, optimization*

INTRODUCTION

I have a family of practical applications that I cannot seem to formulate properly with weighted linear least squares [Claerbout, 1990]. I think I need a wholly new formulation such as the one with eigenvectors that I propose below:

EIGENVECTOR FORMULATION OF MISSING DATA PROBLEM

There are unknowns in a vector $\mathbf{m}$ and knowns in a vector $\mathbf{k}$. Pack $\mathbf{m}$ and $\mathbf{k}$ into a column vector $\mathbf{d}$ which will be known as "data space". There is also a known linear operator $\mathbf{A}$, and I have programs that will rapidly evaluate $\mathbf{Ad}$ and $\mathbf{A'r}$ for any vectors $\mathbf{d}$ and $\mathbf{r}$. Here $\mathbf{A'}$ denotes the conjugate of $\mathbf{A}$.

$$\max_{\mathbf{m}} \ \lambda(\mathbf{m}) \quad = \quad \frac{\begin{bmatrix} \mathbf{m'} & \mathbf{k'} \end{bmatrix} \mathbf{A'A} \begin{bmatrix} \mathbf{m} \\ \mathbf{k} \end{bmatrix}}{\begin{bmatrix} \mathbf{m'} & \mathbf{k'} \end{bmatrix} \begin{bmatrix} \mathbf{m} \\ \mathbf{k} \end{bmatrix}} \quad = \quad \frac{\mathbf{d'A'Ad}}{\mathbf{d'd}} \tag{1}$$

In the physical problem, the components of $\mathbf{m}$ and $\mathbf{k}$ randomly interleave one another and there are huge numbers of components of each. To gain some insight and see how classical mathematics is relevant, forget for a moment the distinction between known and missing data and imagine that all the components of the data field $\mathbf{d}$ are independent variables. Then, it is well known (and I will repeat the proof below) that the stationary values of λ in (1) are the family of eigenvalues. So if components of $\mathbf{k}$ happen to be components of the eigenvector of maximum eigenvalue, then maximization of λ will bring $\mathbf{m}$ to the remaining components of that eigenvector of maximum eigenvalue. My conjecture is that in practice we would find

[1]**email:** not available

that maximization of λ would lead to estimated missing data $\mathbf{m}$ that best fits the various components in the known data $\mathbf{k}$.

In many of our applications, $\mathbf{A}$ is roughly a unitary matrix because it arises from wave propagation problems. Think of λ this way: The denominator limits the total energy; In such cases, maximization should quickly exclude evanescent energy, but a huge number of eigenvalues are all the same, so the maximization is extremely nonunique. To obtain uniqueness with unitary operators we need to introduce a weighting function such as $\lambda(\mathbf{m}) = (\mathbf{d}'\mathbf{A}'\mathbf{W}\mathbf{A}\mathbf{d})/(\mathbf{d}'\mathbf{d})$.

The mathematical problem with the $\mathbf{k}$ constraints however, was unknown to Michael Saunders. Since it is a nonlinear problem we should be wary of multiple unwanted solutions, but that need not stop us from trying. My proposed method for the optimization (1) follows from what I know about conjugate gradients: First we easily find a gradient and keep a vector representing the previous step. Second, scaling these by α and β respectively, $\lambda(\alpha, \beta)$ is a simple ratio of quadratic forms in two variables. After getting the coefficients of these quadratic forms the problem is of such low dimensionality (two) that almost any method should suffice.

GRADIENT

The gradient of λ will assist us in finding the maximum. First recall the derivative of a quotient:

$$\lambda \quad = \quad \frac{n}{d} \tag{2}$$

$$\frac{d\lambda}{dx} \quad = \quad \frac{n_x}{d} \; - \; \frac{n}{d^2}\, d_x \quad = \quad \frac{1}{d}\; (n_x - \lambda d_x) \tag{3}$$

Using (3) we set to zero the gradient of the quadratic form $\lambda(\mathbf{d}) = (\mathbf{d}'\mathbf{A}'\mathbf{A}\mathbf{d})/(\mathbf{d}'\mathbf{d})$ getting $\mathbf{A}'\mathbf{A}\mathbf{d} = \lambda\mathbf{d}$ which proves the eigenvector assertion I made above.

Now restate the gradient while distinguishing missing data from known data and taking the gradient only with respect to the missing. To exhibit this most clearly, we write the $\mathbf{A}$ operator with its columns separated into two parts, one for missing data and one for known, $\mathbf{A} = [\mathbf{B}\mathbf{C}]$

$$\frac{\partial\lambda}{\partial\mathbf{m}'} \quad = \quad \mathbf{B}'\,[\mathbf{B}\mathbf{C}]\begin{bmatrix}\mathbf{m}\\ \mathbf{k}\end{bmatrix} \; - \; \lambda\mathbf{m} \tag{4}$$

Another expression for the gradient offers more insight and is often more practical during coding: Let $\mathbf{M}$ be a square matrix with ones and zeros on the main diagonal where ones are found at locations of missing data and zeros otherwise. Using $\mathbf{M}$, the gradient is

$$\mathbf{g} \quad = \quad \mathbf{M}\left[\mathbf{A}'\mathbf{A}\mathbf{d} \; - \; \frac{\|\,\mathbf{A}\mathbf{d}\,\|}{\|\,\mathbf{d}\,\|}\;\mathbf{d}\right] \tag{5}$$

CODING

Finally, I introduce the specialized notation I like for optimization manipulations. First, I omit bold on vectors. Second, when a vector is transformed by the operator $\mathbf{A}$, I denote the transformed vector by an upper-case letter. Thus $D = \mathbf{A}d$ and $G = \mathbf{A}g$. Let the scalar α denote the distance moved along the gradient. In this notation, perturbations of λ are

$$\lambda(\alpha) \quad = \quad \frac{(D+\alpha G)'(D+\alpha G)}{(d+\alpha g)'(d+\alpha g)} \tag{6}$$

A steepest descent method amounts to:

1. Find the gradient $\mathbf{g}$ using (5)
2. Compute $D = \mathbf{A}d$ and $G = \mathbf{A}g$
3. Maximize the ratio of scalars in (6) by any crude method such as interval division.
4. Repeat.

A conjugate-gradient-like method is like the steepest descent method supplementing the gradient by another vector, the vector of the previous step. Michael Saunders suggested the Pollack-Ribier (sp) method and gave me a Fortran subroutine for the line search.

REFERENCES

Claerbout, J.F., 1990, Extending a CMP gather by optimization in velocity space: SEP–**67**, 43–49.